Universal Approximation of a Class of Interval Type-2 Fuzzy Neural Networks in Nonlinear Identification

Oscar Castillo, 1 Juan R. Castro, 2 Patricia Melin, 1 and Antonio Rodriguez-Diaz 2

1 Tijuana Institute of Technology, 22379 Tijuana, BCN, Mexico
2 Baja California Autonomous University (UABC), 22379 Tijuana, BCN, Mexico

Correspondence should be addressed to Oscar Castillo; ocastillo@hafsamx.org

Received 15 January 2013; Revised 20 June 2013; Accepted 20 June 2013

Copyright © 2013 Oscar Castillo et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Neural networks (NNs), type-1 fuzzy logic systems (T1FLSs), and interval type-2 fuzzy logic systems (IT2FLSs) have been shown to be universal approximators, which means that they can approximate any nonlinear continuous function. Recent research shows that embedding an IT2FLS on an NN can be very effective for a wide number of nonlinear complex systems, especially when handling imperfect or incomplete information. In this paper we show, based on the Stone-Weierstrass theorem, that an interval type-2 fuzzy neural network (IT2FNN) is a universal approximator, which uses a set of rules and interval type-2 membership functions (IT2MFs) for this purpose. Simulation results of nonlinear function identification using the IT2FNN for one and three variables and for the Mackey-Glass chaotic time series prediction are presented to illustrate the concept of universal approximation.

1. Introduction

Several authors have contributed to universal approximation results. An overview can be found in [1–8]; further references to prime contributors in function approximations by neural networks are in [4, 9–12] and type-2 fuzzy logic modeling in [13–23]. It has been shown that a three-layer NN can approximate any real continuous function [24]. The same has been shown for a TIFLS [1, 25] using the Stone-Weierstrass theorem [3]. A similar analysis was made by Kosko [2, 9] using the concept of fuzzy regions. In [3, 26] Buckley shows that, with a Sugeno model [27], a TIFLS can be built with the ability to approximate any nonlinear continuous function. Also, combining the neural and fuzzy logic paradigms [28, 29], an effective tool can be created for approximating any nonlinear function [4]. In this sense, an expert can use a type-1 fuzzy neural network (T1FNN) [10–12, 30] or IT2FNN systems and find interpretable solutions [15–17, 31–34]. In general, Takagi-Sugeno-Kang (TSK) TIFLSs are able to approximate by the use of polynomial consequent rules [7, 27]. This paper uses the Levenberg-Marquardt backpropagation learning algorithm for adapting antecedent and consequent parameters for an adaptive IT2FNN, since its efficiency and soundness characteristics make them fit for these optimizing problems. An Adaptive IT2FNN is used as a universal approximator of any nonlinear functions. A set $\Psi$ of IT2FNNs is universal if and only if (iff), given any process $\Omega$, there is a system $\Phi \in \Psi$ such that the difference between the output from IT2FNN and output from $\Omega$ is less than a given $\epsilon$.

In this paper the main contribution is the proposed IT2FNNs architectures, which are shown to be universal approximators and are illustrated with several benchmark problems to verify their applicability for real world problems.

2. Interval Type-2 Fuzzy Neural Networks

An IT2FNN [15, 31, 35] combines a TSK interval type-2 fuzzy inference system (TSKIT2FIS) [13, 14, 33, 34] with an adaptive NN in order to take advantage of both models best characteristics. In general, when representing IT2FNN graphically, rectangles are used to represent adaptive nodes and circles to represent nonadaptive nodes. Output values of pair nodes (green color) and odd nodes (blue color) represent uncertainty intervals (Figures 1–4). In this kind of interval
type-2 neurofuzzy adaptive networks, nodes represent processing units called neurons, which can be classified into crisp and fuzzy neurons.

The IT2FNN-1 architecture has 5 layers (Figure 1) [35] and consists of adaptive nodes with equivalent function to lower-upper membership in fuzzification layer (layer 1). Nonadaptive nodes in rules layer (layer 2) interconnect with the fuzzification layer (layer 1) in order to generate TSK IT2FIS rules antecedents. Adaptive nodes in the consequent layer (layer 3) are connected to the input layer (layer 0) to generate rules consequents. Nonadaptive nodes in type-reduction layer (layer 4) evaluate left-right values with the Karnik and Mendel (KM) [13,14] algorithm. The nonadaptive nodes in the defuzzification layer (layer 5) average left-right values.

The IT2FNN-3 architecture has 8 layers (Figure 2) [35] and uses IT2FN for fuzzifying the inputs (layers 1-2). Nonadaptive nodes in rules layer (layer 3) interconnect with lower-upper linguistic values layer (layer 2) to generate TSK IT2FIS rules antecedents. Adaptive nodes in layer 4 adapt left-right firing strength, biasing rules lower-upper trigger forces with synaptic weights between layers 3 and 4. Layer 5's nonadaptive nodes normalize rules lower-upper firing strength. Nonadaptive nodes in consequent layer (layer 6) interconnect with input layer (layer 0) to generate rules consequents. Nonadaptive nodes in type-reduction layer (layer 7) evaluate left-right values adding lower-upper product of lower-upper triggering forces normalized by rules consequent left-right values. Node in defuzzification layer is adaptive and its output \( \hat{y} \) is defined as biased average of left-right values and parameter \( y \). Parameter \( y \) (0.5 by default) adjusts uncertainty interval defined by left-right values \([\hat{y}_l, \hat{y}_r]\).

Architectures IT2FNN-0 and IT2FNN-2, which will be shown in Sections 3.2 and 3.3, respectively, as universal approximators, are described with more details in Section 2.1.

2.1. IT2FNN-0 Architecture. An IT2FNN-0 is a seven-layer IT2FNN, which integrates a first order TSK IT2FIS (interval type-2 fuzzy antecedents and real consequents) with an adaptive NN. The IT2FNN-0 (Figure 3) layers are described as follows.

**Layer 0. Inputs**

\[
\alpha_i^0 = x_i, \quad i = 1, \ldots, n. \tag{1}
\]

**Layer 1. Adaptive type-1 fuzzy neuron (T1FN)**

\[
\text{net}_i^1 = w_{k_i}^1 x_i + b_k^1 \quad \forall i = 1, \ldots, n, \quad k = 1, \ldots, B, \tag{2}
\]

\[
\alpha_k^1 = \mu(\text{net}_k^1), \quad \text{where the transfer function } \mu \text{ is a membership function}, \quad \text{net}_k^1 \text{ is the weighted sum of inputs } (x_i) \text{ and}
\]
the synaptic weights ($w_{k,i}^j$), and $b_k^l$ is the threshold for each neuron.

**Layer 2.** Nonadaptive T1FN. This layer contains T-norm and S-norm fuzzy nodes

$$o_{2k-1}^2 = o_{2k-1}^1 \cdot o_{2k}^1 \quad \forall k = 1, \ldots, \theta, \quad T\text{-norm fuzzy node},$$

where $\theta$ is the number of nodes in layers 1 and 2

$$o_{2k}^2 = o_{2k}^1 \cdot o_{2k}^1 - o_{2k-1}^2, \quad S\text{-norm fuzzy node},$$

$\tau = \ell_{k,i}$ for all $k = 1, \ldots, M$, and $i = 1, \ldots, n$, where $\ell_{k,i}$ is the table of indices of the antecedents of the rules $\pi = \sum_{i=1}^{n-1} \nu |\tau|$, where $\pi$ is a vector of indices for each node of layer 2

if $\tau > 0$

$$\mu_k^1 = o_{1i}^2(\pi), \quad \bar{\mu}_k^1 = o_{1i}^2(\pi)$$

else

$$\mu_{k,j} = \text{null}, \quad \bar{\mu}_{k,j} = \text{null}$$

end

where $\mu_k^1, \bar{\mu}_k^1$ are lower and upper membership function values, respectively. $i_1(\pi)$ and $i_2(\pi)$ are vectors with even and odd indices of the nodes of layer 2.

**Layer 3.** Lower-upper firing strength ($\underline{w}^k, \bar{w}^k$). Having nonadaptive nodes for generating lower-upper firing strength of TSK IT2FIS rules (7),

$$o_{2k-1}^3 = \underline{w}^k, \quad o_{2k}^3 = \bar{w}^k,$$

$$\underline{w}^k = \prod_{i=1}^{n} \mu_{\ell_{k,i}}^1(x),$$

$$\bar{w}^k = \prod_{i=1}^{n} \bar{\mu}_{\ell_{k,i}}^1(x),$$

where $\mu_{\ell_{k,i}}^1(x) \in [\mu_{\ell_{k,i}}^1(x), \bar{\mu}_{\ell_{k,i}}^1(x)]$ is the Gaussian interval type-2 membership function, igaussmtype2 ($x, [\mu_k^1 m_k^1, 2m_k^1]$), defined by

$$\mu_{\ell_{k,i}}^1(x) \in [\mu_{\ell_{k,i}}^1(x), \bar{\mu}_{\ell_{k,i}}^1(x)] = \exp \left[-\frac{1}{2} \left( \frac{x_i - m_k^1}{\sigma_k^1} \right)^2 \right], \quad (8)$$
Layer 4. Lower-upper firing strength rule normalization \((\phi^k, \phi^k)\). Nodes in this layer are nonadaptive and the output is defined as the ratio between the \(k\)th lower-upper firing strength rule \((w^k, \bar{w}^k)\) and the sum of lower-upper firing strength of all rules (13) and (14):

\[
\phi^k = \frac{w^k}{D_l}, \quad k = 1, \ldots, M, \tag{13}
\]

where \(D_l = \sum_{k=1}^{M} w^k\),

\[
\bar{\phi}^k = \frac{\bar{w}^k}{D_r}, \quad k = 1, \ldots, M, \tag{14}
\]

where \(D_r = \sum_{k=1}^{M} \bar{w}^k\).

Layer 5. Rule consequents. Each node is adaptive and its parameters are \(\{\xi^k, \xi^k\}\). The node's output corresponds to partial output of \(k\)th rule \(y^k\) (16):

\[
\phi^k = \frac{w^k}{D_l}, \quad k = 1, \ldots, M, \tag{13}
\]

If we view \(\phi^k, \bar{\phi}^k\) as fuzzy basis functions (FBF) (32) and (33) and \(y^k(x)\) as linear function (16), then \(\hat{y}(x)\) can be viewed as a linear combination of the basis functions (20) and (21):

\[
\phi^k = \frac{w^k}{D_l}, \quad k = 1, \ldots, M, \tag{13}
\]

where \(D_l = \sum_{k=1}^{M} w^k\),

\[
\bar{\phi}^k = \frac{\bar{w}^k}{D_r}, \quad k = 1, \ldots, M, \tag{14}
\]

where \(D_r = \sum_{k=1}^{M} \bar{w}^k\).

Layer 5. Rule consequents. Each node is adaptive and its parameters are \(\{\xi^k, \xi^k\}\). The node's output corresponds to partial output of \(k\)th rule \(y^k\) (16):

\[
\phi^k = \frac{w^k}{D_l}, \quad k = 1, \ldots, M, \tag{13}
\]

where \(D_l = \sum_{k=1}^{M} w^k\),

\[
\bar{\phi}^k = \frac{\bar{w}^k}{D_r}, \quad k = 1, \ldots, M, \tag{14}
\]

where \(D_r = \sum_{k=1}^{M} \bar{w}^k\).
Layer 6. Estimating left-right interval values \([\hat{y}_l, \hat{y}_r]\) (18), nodes are nonadaptive with outputs \(\hat{y}_l, \hat{y}_r\). Layer 6 output is defined by

\[
o_1^6 = \hat{y}_l(x), \quad n_2^6 = \hat{y}_r(x),
\]

(17)

where

\[
\hat{y}_l(x) = \sum_{k=1}^{M} \phi_{k}^l(y^k), \quad \hat{y}_r(x) = \sum_{k=1}^{M} \phi_{k}^r(y^k).
\]

(18)

Layer 7. Defuzzification. This layer’s node is adaptive, where the output \(\hat{y}, (20)\) and (21), is defined as weighted average of left-right values and parameter \(\gamma\). Parameter \(\gamma\) (default value 0.5) adjusts the uncertainty interval defined by left-right values \([\hat{y}_l, \hat{y}_r]\):

\[
o_1^7 = \hat{y}(x),
\]

(19)

where

\[
\hat{y}(x) = \gamma \hat{y}_l(x) + (1 - \gamma) \hat{y}_r(x),
\]

(20)

\[
\hat{y}(x) = \sum_{k=1}^{M} \left[ \gamma \phi_{k}^l(x) + (1 - \gamma) \phi_{k}^r(x) \right] y^k(x).
\]

(21)

2.2. IT2FNN-2 Architecture. An IT2FNN-2 [31] is a six-layer IT2FNN, which integrates a first order TSKIT2FIS (interval type-2 fuzzy antecedents and interval type-1 fuzzy consequents), with an adaptive NN. The IT2FNN-2 (Figure 4) layers are described in a similar way to the previous architectures.
3. IT2FNN as a Universal Approximator

Based on the description of the interval type-2 fuzzy neural networks, it is possible to prove that under certain conditions, the resulting IT2FIS has unlimited approximation power to match any nonlinear functions on a compact set [36, 37] using the Stone-Weierstrass theorem [5, 6, 10, 30].

3.1. Stone-Weierstrass Theorem

Theorem 1 (Stone-Weierstrass theorem). Let \( Z \) be a set of real continuous functions on a compact set \( U \). If (1) \( Z \) is an algebra, that is, the set \( Z \) is closed under addition, multiplication, and scalar multiplication, (2) \( Z \) separates points on \( U \), that is, for every \( x, y \in U \), \( x \neq y \), there exists \( f \in Z \) such that \( f(x) \neq f(y) \), and (3) \( Z \) vanishes at no point of \( U \), that is, for each \( x \in U \) there exists \( f \in Z \) such that \( f(x) = 0 \), then the uniform closure of \( Z \) consists of all real continuous functions on \( U \); that is, \((Z, d_{\infty})\) is dense in \((C[U], d_{\infty})\) [36–38].

Theorem 2 (universal approximation theorem). For any given real continuous function \( g(u) \) on the compact set \( U \subset \mathbb{R}^n \) and arbitrary \( \varepsilon > 0 \), there exists \( f \in Z \) such that \( \sup_{x \in U} |g(x) - f(x)| < \varepsilon \).

3.2. Applying Stone-Weierstrass Theorem to the IT2FNN-0 Architecture. In the IT2FNN-0, the domain on which we operate is almost always compact. It is a standard result in real analysis that every closed and bounded set in \( \mathbb{R}^n \) is compact. Now we shall apply the Stone-Weierstrass theorem to show the representational power of IT2FNN with simplified fuzzy if-then rules. We now consider a subset of the IT2FNN-0 on Figure 5. The set of IT2FNN-0 with singleton fuzzifier, product inference, center of sets type reduction, and Gaussian interval type-2 membership function consists of all FBF expansion functions of the form (38), (40). \( f : U \subset \mathbb{R}^n \to \mathbb{R} \), \( x = (x_1, x_2, \ldots, x_n) \in U \); \( \mu_{\tilde{F}}(x) \in [\mu_{\tilde{F}_i}(x), \mu_{\tilde{F}_i}(x)] \) is the Gaussian interval type-2 membership function, igaussstype2 \( (x, [\sigma_i^k, m_i^k, m_i^k]) \), defined by (27) and (31). If we view \( \phi^k(x), \tilde{\phi}^k(x) \) as fuzzy basis functions (32) and (33) and \( y^k(x) \) are functionals (34), then \( \tilde{y}(x) \) of (38) and (40) can be viewed as a linear combination of the fuzzy basis functions, and then the IT2FNN-0 system is equivalent to an FBF expansion. Let \( Y \) be the set of all the FBF expansions (38) and (40) with \( \phi^k(x), \tilde{\phi}^k(x) \) given by (13) and (38) and let \( d_{\infty}(f_1, f_2) = \sup_{x \in U} |f_1(x) - f_2(x)| \) be the sup metric; then, \((Y, d_{\infty})\) is a metric space [38]. We use the following Stone-Weierstrass theorem to prove our result.

Suppose we have two IT2FNN-0s \( f_1, f_2 \in Y \); the output of each system can be expressed as

\[
f_1(x) = \alpha \tilde{y}_1^1(x) + (1 - \alpha) \tilde{y}_1^2(x),
\]

where

\[
\tilde{y}_1^1(x) = \sum_{k=1}^{M} \phi^k_1(x) z^k_1(x) = \sum_{k=1}^{M} \frac{w^k_1(x) z^k_1(x)}{D^1_f},
\]

\[
\tilde{y}_1^2(x) = \sum_{k=1}^{M} \phi^k_2(x) z^k_2(x) = \sum_{k=1}^{M} \frac{w^k_2(x) z^k_2(x)}{D^2_f},
\]

where

\[
D^1_f = \prod_{k=1}^{M} \prod_{i=1}^{n} \mu^k_{\tilde{F}_i}(x), \quad D^2_f = \prod_{k=1}^{M} \prod_{i=1}^{n} \overline{\mu}_{\tilde{F}_i}(x),
\]

\[
w^k_1 = \prod_{i=1}^{n} \mu^k_{\tilde{F}_i}(x), \quad w^k_2 = \prod_{i=1}^{n} \overline{\mu}_{\tilde{F}_i}(x),
\]

\[
\tilde{y}_2^1(x) = y_2^1(x) + \gamma \tilde{y}_2^2(x),
\]

where

\[
\tilde{y}_2^1(x) = \sum_{k=1}^{M} \phi^k_1(x) z^k_1(x) = \sum_{k=1}^{M} \frac{w^k_1(x) z^k_1(x)}{D^1_f},
\]

\[
\tilde{y}_2^2(x) = \sum_{k=1}^{M} \phi^k_2(x) z^k_2(x) = \sum_{k=1}^{M} \frac{w^k_2(x) z^k_2(x)}{D^2_f},
\]

where

\[
D^2_f = \prod_{k=1}^{M} \prod_{i=1}^{n} \mu^k_{\tilde{F}_i}(x), \quad D^2_f = \prod_{k=1}^{M} \prod_{i=1}^{n} \overline{\mu}_{\tilde{F}_i}(x),
\]

\[
w^k_1 = \prod_{i=1}^{n} \mu^k_{\tilde{F}_i}(x), \quad w^k_2 = \prod_{i=1}^{n} \overline{\mu}_{\tilde{F}_i}(x),
\]

\[
\tilde{y}_2^1(x) = y_2^1(x) + \gamma \tilde{y}_2^2(x),
\]

Lemma 3. \( Y \) is closed under addition.

Proof. The proof of this lemma requires our IT2FNN-0 to be able to approximate sums of functions. Suppose we have
two IT2FNN-0s, $f_1(x)$ and $f_2(x)$ with $M_1$ and $M_2$ rules, respectively. The output of each system can be expressed as

$$f_1(x) + f_2(x) = \sum_{k_1=1}^{M_1} \sum_{k_2=1}^{M_2} \left[ \frac{w_{k_1}^1 w_{k_2}^2 (\alpha z_{k_1}^1 + \gamma z_{k_2}^1)}{D_1^1 D_1^2} \right] + \sum_{k_1=1}^{M_1} \sum_{k_2=1}^{M_2} \left[ \frac{w_{k_1}^1 w_{k_2}^2 \{(1-\alpha)z_{k_1}^1 + (1-\gamma)z_{k_2}^1\}}{D_1^1 D_2^2} \right]$$

(29)

and that $\Phi_{k_1,k_2} = (w_{k_1}^1 w_{k_2}^2)/(D_1^1 D_1^2)$ and $\overline{\Phi}_{k_1,k_2} = (w_{k_1}^1 w_{k_2}^2)/(D_1^1 D_2^2)$, where the FBFs are known to be nonlinear. Therefore, an equivalent to IT2FNN-0 can be constructed under the addition of $f_1(x)$ and $f_2(x)$, where the consequents form an addition of $\alpha z_{k_1}^1 + \gamma z_{k_2}^1$ and $(1-\alpha)z_{k_1}^1 + (1-\gamma)z_{k_2}^1$, multiplied by a respective FBFs expansion (Theorem 1), and there exists $f \in \mathcal{Y}$ such that $\sup_{x \in U} |g(x) - f(x)| < \varepsilon$ (Theorem 2). Since $f(x)$ satisfies Lemma 3 and $\mathcal{Y} \in \mathcal{Y}(x) = f_1(x) + f_2(x)$ then we can conclude that $\mathcal{Y}$ is closed under addition. Note that $z_{k_1}^1$ and $z_{k_2}^1$ can be linear since the FBFs are a nonlinear basis interval and therefore the resultant function, $f(x)$, is nonlinear interval (see Figure 5).

**Lemma 4.** $\mathcal{Y}$ is closed under multiplication.

**Proof.** Similar to Lemma 3, we model the product of $f_1(x)f_2(x)$ of two IT2FNN-0s. The product $f_1(x)f_2(x)$ can be expressed as

$$f_1(x)f_2(x) = \frac{1}{D_1^1 D_2^2} \left[ \sum_{k_1=1}^{M_1} \sum_{k_2=1}^{M_2} w_{k_1}^1 w_{k_2}^2 \alpha z_{k_1}^1 z_{k_2}^1 \right] + \frac{1}{D_1^1 D_2^2} \left[ \sum_{k_1=1}^{M_1} \sum_{k_2=1}^{M_2} w_{k_1}^1 w_{k_2}^2 \gamma z_{k_1}^1 z_{k_2}^1 \right] + \frac{1}{D_1^1 D_2^2} \left[ \sum_{k_1=1}^{M_1} \sum_{k_2=1}^{M_2} w_{k_1}^1 w_{k_2}^2 (1-\alpha) \gamma z_{k_1}^1 z_{k_2}^1 \right] + \frac{1}{D_1^1 D_2^2} \left[ \sum_{k_1=1}^{M_1} \sum_{k_2=1}^{M_2} w_{k_1}^1 w_{k_2}^2 (1-\gamma) \alpha z_{k_1}^1 z_{k_2}^1 \right].$$

(30)
Therefore, an equivalent to IT2FNN-0 can be constructed under the multiplication of $f_1(x)$ and $f_2(x)$, where the consequents form an addition of $a \gamma z_{i1}^1 z_{i2}^1, a (1 - \gamma) z_{i1}^1 z_{i2}^2, (1 - \alpha) \gamma z_{i1}^2 z_{i2}^2,$ and $(1 - \alpha) (1 - \gamma) z_{i1}^2 z_{i2}^2$ multiplied by a respective FBFs $\Phi_{k1, k2} = (w_{k1}^1 w_{k2}^2)/(D_1^1 D_2^2)$, $\Phi_{k1, k2} = (w_{k1}^2 w_{k2}^1)/(D_1^2 D_2^1)$, $\Phi_{k1, k2} = (w_{k1}^1 w_{k2}^2)/(D_1^1 D_2^2)$, and $\Phi_{k1, k2} = (w_{k1}^2 w_{k2}^1)/(D_1^2 D_2^1)$ expansion (Theorem 1), and there exists $f \in Y$ such that $sup_{x\in\Xi} (g(x) - f(x)) < \epsilon$ (Theorem 2). Since $f(x)$ satisfies Lemma 3 and $Y \in f(x) = f_1(x) f_2(x)$ then we can conclude that $Y$ is closed under multiplication. Note that $z_{i1}^1$ and $z_{i2}^2$ can be linear since the FBFs are a nonlinear basis interval and therefore the resultant function, $f(x)$, is nonlinear interval. Also, even if $z_{i1}^1$ and $z_{i2}^2$ were linear, their product $z_{i1}^1 z_{i2}^2$ is evidently polynomial (see Figure 5).

Lemma 5. Y is closed under scalar multiplication.

Proof. Let an arbitrary IT2FNN-0 be $f(x)$ (20); the scalar multiplication of $c f(x)$ can be expressed as

$$
 cf(x) = c \prod_{i=1}^{n} \frac{a D_i w_{ki}^1(x) + (1 - \alpha) D_i w_{ki}^2(x) c z_{i1}^1 z_{i2}^2(x)}{D_i D_r}.
$$

Therefore we can construct an IT2FNN-0 that computes $c z^2(x)$ in the form of the proposed IT2FNN-0; $Y$ is closed under scalar multiplication.

Lemma 6. For every $x^0, y^0 \in U$ and $x^0 \neq y^0$, there exists $f \in Y$ such that $f(x^0) \neq f(y^0)$; that is, $Y$ separates points on $U$.

Proof. We prove that $(Y, d_{\infty})$ separates points on $U$. We prove this by constructing a required $f(x)$ (20); that is, we specify $f \in Y$ such that $f(x^0) \neq f(y^0)$ for arbitrarily given $(x^0, y^0) \in U$ with $x^0 \neq y^0$. We choose two fuzzy rules in the form of (8) for the fuzzy rule base (i.e., $M = 2$). Let $x^0 = (x_{i1}^0, x_{i2}^0, \ldots, x_n^0)$ and $y^0 = (y_{i1}^0, y_{i2}^0, \ldots, y_n^0)$. If $x_i^0 = (x_{i1} + x_{i2}^0)/2$ and $y_i^0 = (y_{i1} + y_{i2}^0)/2$ with $x_i^0 \neq y_i^0$, we define two interval type-2 fuzzy sets $(\bar{F}^1_i, [\underline{\mu}^1_i, \bar{\mu}^1_i])$ and $(\bar{F}^2_i, [\underline{\mu}^2_i, \bar{\mu}^1_i])$ with

\begin{align*}
\bar{\mu}^1_i(x_i) &= \begin{cases} 
\exp \left[ -\frac{1}{2} (x_i - y_{i1}^0)^2 \right], & x_i < y_{i1}^0, \\
1, & x_{i1}^0 \leq x_i \leq x_{i1}^0, \\
\exp \left[ -\frac{1}{2} (x_i - x_{i2}^0)^2 \right], & x_i > x_{i2}^0,
\end{cases} \\
\underline{\mu}^1_i(x_i) &= \begin{cases} 
\exp \left[ -\frac{1}{2} (x_i - x_{i1}^0)^2 \right], & x_i \leq x_{i1}^0, \\
\exp \left[ \frac{1}{2} (x_i - x_{i2}^0)^2 \right], & x_i > x_{i1}^0,
\end{cases}
\end{align*}

\begin{align*}
\bar{\mu}^2_i(x_i) &= \begin{cases} 
\exp \left[ -\frac{1}{2} (x_i - y_{i2}^0)^2 \right], & x_i < y_{i2}^0, \\
1, & y_{i2}^0 \leq x_i \leq y_{i2}^0, \\
\exp \left[ -\frac{1}{2} (x_i - x_{i1}^0)^2 \right], & x_i > x_{i1}^0,
\end{cases} \\
\underline{\mu}^2_i(x_i) &= \begin{cases} 
\exp \left[ -\frac{1}{2} (x_i - x_{i1}^0)^2 \right], & x_i \leq x_{i2}^0, \\
\exp \left[ \frac{1}{2} (x_i - x_{i1}^0)^2 \right], & x_i > x_{i1}^0,
\end{cases}
\end{align*}

\begin{align*}
\bar{\phi}^1(y) &= \frac{\prod_{i=1}^{n} \bar{\mu}^1_i(y_i^0)}{\prod_{i=1}^{n} \underline{\mu}^1_i(y_i^0)}, \\
\underline{\phi}^1(y) &= \frac{1}{\prod_{i=1}^{n} \bar{\mu}^1_i(y_i^0) + \prod_{i=1}^{n} \underline{\mu}^1_i(y_i^0)},
\end{align*}

\begin{align*}
\bar{\phi}^2(y) &= \frac{\prod_{i=1}^{n} \bar{\mu}^2_i(y_i^0)}{\prod_{i=1}^{n} \underline{\mu}^2_i(y_i^0)}, \\
\underline{\phi}^2(y) &= \frac{1}{\prod_{i=1}^{n} \bar{\mu}^2_i(y_i^0) + \prod_{i=1}^{n} \underline{\mu}^2_i(y_i^0)}.
\end{align*}

Since $x^0 \neq y^0$, there must be some $i$ such that $x_i^0 = y_i^0$; hence, we have $\prod_{i=1}^{n} \bar{\mu}^1_i(y_i^0) \neq 1$ and $\prod_{i=1}^{n} \bar{\mu}^2_i(y_i^0) \neq 1$. If we choose
\[ z^1 = 0 \text{ and } z^2 = 1, \text{ then } f(x^0) = \alpha \phi_{\gamma}^2(x^0) + (1-\alpha)[1 - \phi_{\text{w}}^1(x^0)] \neq \alpha \phi_{\gamma}^2(y^0) + (1 - \alpha)\phi_{\text{w}}^1(y^0) = f(y^0). \] Separability is satisfied whenever an IT2FNN-0 can compute strictly monotonic functions of each input variable. This can easily be achieved by adjusting the membership functions of the premise part. Therefore, \((Y, d_{\text{co}})\) separates points on \(U\).

**Lemma 7.** For each \(x \in U\), there exists \(f \in Y\) such that \(f(x) \neq 0\); that is, \(Y\) vanishes at no point of \(U\).

Finally, we prove that \((Y, d_{\text{co}})\) vanishes at no point of \(U\). By observing (8)–(11), (20), and (21), we simply choose all \(y_k^j(x) > 0 \) \((k = 1, 2, \ldots, M)\); that is, any \(f \in Y\) with \(y_k^j(x) > 0\) serves as the required \(f\).

**Proof of Theorem 2.** From (20) and (21), it is evident that \(Y\) is a set of real continuous functions on \(U\), which are established by using complete interval type-2 fuzzy sets in the IF parts of fuzzy rules. Using Lemmas 3, 4, and 5, \(Y\) is proved to be an algebra. By using the Stone-Weierstrass theorem together with Lemmas 6 and 7, we establish that the proposed IT2FNN-0 possesses the universal approximation capability.

### 3.3. Applying the Stone-Weierstrass Theorem to the IT2FNN-2 Architecture

We now consider a subset of the IT2FNN-2 on Figure 2. The set of IT2FNN-2 with singleton fuzzifier, product inference, type-reduction defuzzifier (KM) [13, 14], and Gaussian interval type-2 membership function consists of all FBF expansion functions. \(f: U \subseteq R^n \rightarrow R, x = (x_1, x_2, \ldots, x_n) \in U; \mu_{\beta_i}^j(x) \in ([\mu_{\beta_i}^j(x), \overline{\mu}_{\beta_i}^j(x)])\) is the Gaussian interval type-2 membership function, igauss type2 \((x, [\sigma_j^k, 1, m_m^k, 1, m_1^k]),\) defined by (8)–(11). If we view \(\phi_{\gamma}^j(x), \phi_{\text{w}}^k(x), \phi_{\text{w}}^j(x), \phi_{\text{w}}^k(x)\) as basis functions (44), (46), (49), (50) and \(y_k^j(x), y_k^k(x)\) are linear functions (41), then \(y(x)\) can be viewed as a linear combination of the basis functions. Let \(Y\) be the set of all the FBF expansions with \(\phi_{\gamma}^j(x), \phi_{\text{w}}^k(x), \phi_{\text{w}}^j(x), \phi_{\text{w}}^k(x)\) and let \(d_{\text{co}}(f_1, f_2) = \sup_{x \in U}(|f_1(x) - f_2(x)|)\) be the supmetric; then, \((Y, d_{\text{co}})\) is a metric space [38]. The following theorem shows that \((Y, d_{\text{co}})\) is dense in \((C(U), d_{\text{co}})\), where \(C(U)\) is the set of all real continuous functions defined on \(U\). We use the following Stone-Weierstrass theorem to prove the theorem.

Suppose we have two IT2FNN-2s \(f_1, f_2 \in Y\); the output of each system can be expressed as
\[
\tilde{y}_i^f(x) = \sum_{k=1}^{m_1} \frac{\sum_{l=1}^{L_1} \tilde{\mu}_{\text{w}}^l(x) \tilde{\phi}_l^k(x) \tilde{z}_l^k(x)}{D_i^f} + \sum_{k=L_1+1}^{M_1} \tilde{\mu}_{\text{w}}^l(x) \tilde{\phi}_l^k(x) \tilde{z}_l^k(x),
\]
where
\[
\tilde{y}_i^f(x) = \frac{\sum_{k=1}^{m_1} \tilde{\mu}_{\text{w}}^l(x) \tilde{\phi}_l^k(x) \tilde{z}_l^k(x)}{D_i^f} + \frac{\sum_{k=L_1+1}^{M_1} \tilde{\mu}_{\text{w}}^l(x) \tilde{\phi}_l^k(x) \tilde{z}_l^k(x)}{D_i^f},
\]
\[
\tilde{y}^2_f(x) = \frac{\sum_{k=1}^{m_1} \tilde{\mu}_{\text{w}}^l(x) \tilde{\phi}_l^k(x) \tilde{z}_l^k(x)}{D_i^f} + \frac{\sum_{k=L_1+1}^{M_1} \tilde{\mu}_{\text{w}}^l(x) \tilde{\phi}_l^k(x) \tilde{z}_l^k(x)}{D_i^f}.
\]
\[ y^2_r(x) = \sum_{k_1=1}^{R_1} \sum_{k_2=1}^{R_2} w_{k_1}^2(x) z_{k_1}^2(x) + \sum_{k_1=1}^{M_1} \sum_{k_2=L_2}^{R_2} \phi_{k_1}^2(x) \phi_{k_2}^2(x) \]

where

\[ w_{k_1}^2(x) = \prod_{i=1}^{n} \mu_i^{p_{k_1}}(x), \quad \overline{w}_{k_2}^2(x) = \prod_{i=1}^{n} \mu_i^{q_{k_2}}(x), \]

\[ D_1^2 = \sum_{k_1=1}^{L_1} w_{k_1}^2(x) + \sum_{k_1=L_1+1}^{M_1} w_{k_1}^2(x), \quad D_2^2 = \sum_{k_1=1}^{R_2} \overline{w}_{k_2}^2(x) + \sum_{k_1=R_2+1}^{M_2} \overline{w}_{k_2}^2(x), \]

\[ z_{k_1}^2(x) = \frac{w_{k_1}^2(x)}{D_1^2} \quad \forall k_2 = 1, \ldots, L_2(x), \]

\[ 2 \phi_{k_1}^2(x) = \frac{w_{k_1}^2(x)}{D_1^2} \quad \forall k_2 = 1, \ldots, L_2(x), \]

\[ 2 \phi_{k_2}^2(x) = \frac{w_{k_2}^2(x)}{D_1^2} \quad \forall k_2 = 1, \ldots, R_2(x), \]

\[ 2 \phi_{k_2}^2(x) = \frac{w_{k_2}^2(x)}{D_2^2} \quad \forall k_2 = 1, \ldots, R_2(x), \]

\[ 2 z_{k_1}^2(x) = \sum_{i=1}^{n} c_{k_1}^i x_i + z_{k_1}, \quad \forall k_2 = 1, \ldots, M_2, \]

\[ 2 z_{k_2}^2(x) = \sum_{i=1}^{n} c_{k_2}^i x_i + z_{k_2}, \quad \forall k_2 = 1, \ldots, M_2, \]

Lemma 8. \( Y \) is closed under addition.

Proof. The proof of this lemma requires our IT2FNN-2 to be able to approximate sums of functions. Suppose we have two IT2FNN-2s \( f_1(x) \) and \( f_2(x) \) with rules \( M_1 \) and \( M_2 \), respectively. The output of each system can be expressed as

\[ f_1(x) + f_2(x) = \left( \sum_{k_1=1}^{M_1} \sum_{k_2=L_2}^{R_2} \left[ aD_1^2 \overline{w}_{k_1}^2(x) z_{k_1}^2(x) + yD_1^2 w_{k_2}^2(x) \right] \right) \times \left( D_1^2 D_2^2 \right)^{-1} \]

Therefore, an equivalent to IT2FNN-2 can be constructed under the addition of \( f_1(x) \) and \( f_2(x) \), where the consequents form an addition of \( x_1^k + y \cdot x_2^k \) multiplied by a respective FBFs expansion (Theorem 1), and there exists \( f \in Y \) such that \( \sup_{x \in \mathbb{R}} \left| (f(x) - f(x)) \right| < \epsilon \) (Theorem 2). Since \( f(x) \) satisfies Lemma 3 and \( Y \in f(x) = f_1(x) + f_2(x) \), then we can conclude that \( Y \) is closed under addition. Note that \( z_{k_1}^2 \) and \( z_{k_2}^2 \) can be linear interval since the FBFs are a nonlinear basis and therefore the resultant function, \( f(x) \), is nonlinear interval (see Figure 6).

Lemma 9. \( Y \) is closed under multiplication.

Proof. In a similar way to Lemma 8, we model the product of \( f_1(x)f_2(x) \) of two IT2FNN-2s which is the last point we need to demonstrate before we can conclude that the Stone-Weierstrass theorem can be applied to the proposed reasoning mechanism. The product \( f_1(x)f_2(x) \) can be expressed as

\[ f_1(x)f_2(x) = \frac{\alpha y}{D_1^2 D_2^2} \times \left[ \sum_{k_1=1}^{M_1} \sum_{k_2=L_2}^{R_2} \overline{w}_{k_1}^2(x) z_{k_1}^2(x) z_{k_2}^2(x) \right] \]

\[ + \sum_{k_1=1}^{M_1} \sum_{k_2=L_2+1}^{R_2} \overline{w}_{k_1}^2(x) z_{k_1}^2(x) z_{k_2}^2(x) \]

\[ + \sum_{k_1=L_2+1}^{M_2} \sum_{k_2=L_2+1}^{R_2} \overline{w}_{k_1}^2(x) z_{k_1}^2(x) z_{k_2}^2(x) \]
Figure 6: An example of the IT2FNN-2 architecture.
Therefore, an equivalent to IT2FNN-2 can be constructed under the multiplication of \( f_1(x) \) and \( f_2(x) \), where the consequents form an addition of \( \alpha y^1 z_i^{k_i} + z_i^{k_i} \alpha (1 - \gamma) y^{1} z_i^{k_i} + (1 - \gamma) y^{1} z_i^{k_i} \) and \( (1 - \alpha) y^{1} z_i^{k_i} + (1 - \alpha) (1 - \gamma) y^{1} z_i^{k_i} \), multiplied by a respective FFBs expansion (Theorem 1), and there exists f \( \in Y \) such that \( sup_{x \in Y} |g(f(x) - f(x))| < \varepsilon \) (Theorem 2). Since \( f(x) \) satisfies Lemma 3 and \( Y \in f(x) = f_1(x) f_2(x) \) then we can conclude that \( Y \) is closed under multiplication. Note that \( z_i^{k_i} \) and \( z_i^{k_i} \) can be linear intervals since the FFBs are a nonlinear basis interval and therefore the resultant function, \( f(x) \), is nonlinear interval. Also, even if \( z_i^{k_i} \) and \( z_i^{k_i} \) were linear, their product \( z_i^{k_i} \) is evidently polynomial interval (see Figure 10). □

**Lemma 10.** \( Y \) is closed under scalar multiplication.

**Proof.** Let an arbitrary IT2FNN-2 be \( f(x) \) (51); the scalar multiplication of \( cf(x) \) can be expressed as

\[
cf(x) = \alpha cy_0^1 + (1 - \alpha) c n^1_D (x) = \sum_{i=1}^{M_1} \sum_{k_2=1}^{R_2} \tilde{w}_2^{k_2}(x) (1 - \alpha) c^l \tilde{z}_2^{k_2} (x)
\]

\[
+ \sum_{i=1}^{R_1} \sum_{k_1=1}^{M_1} \tilde{w}_1^{k_1}(x) \tilde{w}_2^{k_2}(x) + \sum_{i=1}^{M_1} \sum_{k_1=1}^{R_1} \sum_{k_2=1}^{M_2} \tilde{w}_1^{k_1}(x) \tilde{w}_2^{k_2}(x)
\]

Therefore we can construct an IT2FNN-2 that computes all FFB expansion combinations with \( ac^l z_1^{k_1}(x) \) and \( (1 - \alpha) c^l z_2^{k_1}(x) \) in the form of the proposed IT2FNN-2, and \( Y \) is closed under scalar multiplication. □

**Lemma 11.** For every \( (x_0, y_0) \in U \) and \( x_0 \neq y_0 \), there exists \( f \in Y \) such that \( f(x_0) \neq f(y_0) \); that is, \( Y \) separates points on \( U \).

We prove this by constructing a required \( f \); that is, we specify \( f \in Y \) such that \( f(x_0) \neq f(y_0) \) for arbitrarily given \( x_0, y_0 \in U \) with \( x_0 \neq y_0 \). We choose two fuzzy rules in the form of (8) for the fuzzy rule base (i.e., \( M = 2 \)). Let \( x_0 = (x_0^1, x_0^2, \ldots, x_0^n) \) and \( y_0 = (y_0^1, y_0^2, \ldots, y_0^n) \). If \( x_0^i = (x_0^i + y_0^i)/2 \)

\[
y_0^i = (y_0^i + y_0^i)/2
\]

and \( y_0^i \neq y_0^i \), we define two interval type-2 fuzzy sets \( (\bar{F}_1, [\mu_{\bar{F}_1}], \bar{\mu}_{\bar{F}_1}) \) and \( (\bar{F}_2, [\mu_{\bar{F}_2}], \bar{\mu}_{\bar{F}_2}) \) with

\[
\bar{\mu}_{\bar{F}_1}(x_i) = \begin{cases} 
\frac{1}{2} (x_i - x_0^i)^2, & x_i < x_0^i, \\
1, & x_i^0 \\
\frac{1}{2} (x_i - x_0^i)^2, & x_i > x_0^i,
\end{cases}
\]

\[
\bar{\mu}_{\bar{F}_2}(x_i) = \begin{cases} 
\frac{1}{2} (x_i - y_0^i)^2, & x_i < y_0^i, \\
1, & y_0^i \leq x_i \leq y_0^i, \\
\frac{1}{2} (x_i - y_0^i)^2, & x_i > y_0^i,
\end{cases}
\]

\[
f(y_0^i) = \alpha \left[ \psi_1^1 (y_0^i) \tilde{z}_1^i (y_0^i) + (1 - \psi_1^1 (y_0^i)) \tilde{z}_2^i (y_0^i) \right]
\]

\[
+ \sum_{i=1}^{M_1} \sum_{k_1=1}^{R_1} \tilde{w}_1^{k_1}(x) (1 - \alpha) c^l \tilde{z}_2^{k_1} (x)
\]

\[
+ \sum_{i=1}^{M_1} \sum_{k_1=1}^{R_1} \sum_{k_2=1}^{M_2} \tilde{w}_1^{k_1}(x) \tilde{w}_2^{k_2}(x)
\]

(52)

\[
\psi_1^1 (x_0^i) = \frac{1}{1 + \prod_{i=1}^{n} \bar{\mu}_{\bar{F}_1} (x_0^i)}
\]

\[
\psi_2^1 (x_0^i) = \frac{\prod_{i=1}^{n} \bar{\mu}_{\bar{F}_2} (x_0^i)}{\prod_{i=1}^{n} \bar{\mu}_{\bar{F}_1} (x_0^i) + \prod_{i=1}^{n} \bar{\mu}_{\bar{F}_2} (x_0^i)}
\]

\[
f \left( y_0^i \right) = \alpha \left[ \psi_1^1 (y_0^i) \tilde{z}_1^i (y_0^i) + \psi_2^1 (y_0^i) \tilde{z}_2^i (y_0^i) \right]
\]

\[
\alpha \left[ (1 - \psi_2^1 (y_0^i)) \tilde{z}_1^i (y_0^i) + \psi_2^1 (y_0^i) \tilde{z}_2^i (y_0^i) \right],
\]

(54)

\[
\psi_2^2 (y_0^i) = \frac{1}{\prod_{i=1}^{n} \bar{\mu}_{\bar{F}_1} (y_0^i) + 1}
\]
\[ \phi_1(y^0) = \prod_{i=1}^{n} \mu_{F_1_i}(y^0_i) \]
\[ \phi_2(y^0) = \prod_{i=1}^{n} \mu_{F_2_i}(y^0_i) \]

Since \( x^0 \neq y^0 \), there must be some \( i \) such that \( x^0_i = y^0_i \); hence, we have \( \prod_{i=1}^{n} \mu_{F_1_i}(x^0_i) \neq 1 \) and \( \prod_{i=1}^{n} \mu_{F_2_i}(x^0_i) \neq 1 \). If we choose \( z^k_r = 0 \) and \( z^k_t = 1 \), then \( f(x^0) = \alpha(1 - \phi_1(x^0)) + (1 - \alpha)\phi_2(x^0) \neq \alpha\phi_1(y^0) + (1 - \alpha)\phi_2(y^0) = f(y^0) \). Therefore, \((Y, \delta_{co})\) separates point on \( U \).

**Lemma 12.** For each \( x \in U \), there exists \( f \in Y \) such that \( f(x) \neq 0 \); that is, \( Y \) vanishes at no point of \( U \).

Finally, we prove that \((Y, \delta_{co})\) vanishes at no point of \( U \). By observing (8)–(11), (20), and (21), we just choose all \( z^k > 0 \) \((k = 1, 2, \ldots, M)\); that is, any \( f \in Y \) with \( z^k > 0 \) serves as required \( f \).

**Proof of Theorem 2.** From (20) and (21), it is evident that \( Y \) is a set of real continuous functions on \( U \), which are established by using complete interval type-2 fuzzy sets in the IF parts of fuzzy rules. Using Lemmas 8, 9, and 10, \( Y \) is proved to be an algebra. By using the Stone-Weierstrass theorem together with Lemmas 11 and 12, we establish that the proposed IT2FNN-2 possesses the universal approximation capability.

Therefore by choosing appropriate class of interval type-2 membership functions, we can conclude that the IT2FNN-0 and IT2FNN-2 with simplified fuzzy if-then rules satisfy the five criteria of the Stone-Weierstrass theorem.

**4. Application Examples**

In this section the results from simulations using ANFIS, IT2FNN-0, IT2FNN-1 [35], IT2FNN-2, and IT2FNN-3 [35] are presented for nonlinear system identification and forecasting the Mackey-Glass chaotic time series [39] with \( r = 60 \) with different signal noise ratio values, SNR(db) = 0, 10, 20, 30, free as uncertainty source. These examples are used as benchmark problems to test the proposed ideas in the paper. We have to mention that the IT2FNN-1 and IT2FNN-3 architectures are very similar to IT2FNN-0 and IT2FNN-2, respectively [35], and their results are presented for comparison purposes. The proposed IT2FNN architectures are validated using 10-fold cross-validation [40, 41] considering sum of square errors (SSE) or root mean square error (RMSE) in the training or test phase. We use cross-validation to measure the variability of the RMSE in the training and testing phases to compare network architectures IT2FNN. Cross-validation procedure evaluation is done using Matlab's `crossvalind` function. Noise is added by Matlab's `awgn` function.

In \( K \)-fold cross-validation [40], the original sample is randomly partitioned into \( K \) subsamples. Of the \( K \) subsamples, a single subsample is retained as the validation data for testing the model, and the remaining \( K - 1 \) subsamples are used as training data. The cross-validation process is then repeated \( K \) times (the folds), with each of the \( K \) subsamples used exactly once as the validation data. The \( K \) results from the folds then can be averaged (or otherwise combined) to produce a single estimation. The advantage of this method over repeated random subsampling is that all observations are used for both training and validation, and each observation is used for validation exactly once. 10-fold cross-validation is commonly used. Three application examples are used to illustrate proofs of universality, as follows.

**Experiment 1** (identification of a one variable nonlinear function). In this experiment we approximate a nonlinear function \( f : \mathcal{R} \rightarrow \mathcal{R} \):

\[ f(u) = 0.6 \sin(\pi u) + 0.3 \sin(3\pi u) + 0.1 \sin(5\pi u) + \eta, \quad (56) \]

(where \( \eta \) is a uniform noise component) using a one input one-output IT2FNN, 50 training data sets with 10-fold cross-validation with uniform noise levels, six IT2MFs type igaussmtype2, 6 rules, and 50 epochs. Once the ANFIS and IT2FNN models are identified a comparison was made, taking into account RMSE statistic values with 10-fold cross-validation. Table 1 and Figure 7 show the resulting RMSE.
Table 2: Resulting RMSE (CHK) values in ANFIS and IT2FNN for non-linearity identification in Experiment 2 with 10-fold cross-validation.

<table>
<thead>
<tr>
<th>SNR (dB)</th>
<th>ANFIS</th>
<th>IT2FNN-0</th>
<th>IT2FNN-1</th>
<th>IT2FNN-2</th>
<th>IT2FNN-3</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1.0432</td>
<td>0.7203</td>
<td>0.6523</td>
<td>0.5512</td>
<td>0.5267</td>
</tr>
<tr>
<td>10</td>
<td>0.3096</td>
<td>0.2798</td>
<td>0.2583</td>
<td>0.2464</td>
<td>0.2344</td>
</tr>
<tr>
<td>20</td>
<td>0.1703</td>
<td>0.1637</td>
<td>0.1592</td>
<td>0.1465</td>
<td>0.1387</td>
</tr>
<tr>
<td>30</td>
<td>0.1526</td>
<td>0.1408</td>
<td>0.1368</td>
<td>0.1323</td>
<td>0.1312</td>
</tr>
<tr>
<td>Free</td>
<td>0.1503</td>
<td>0.1390</td>
<td>0.1323</td>
<td>0.1304</td>
<td>0.1276</td>
</tr>
</tbody>
</table>

Experiment 2 (identification of a three variable nonlinear function). A three-input one-output IT2FNN is used to approximate nonlinear Sugeno [27] function $f: \mathbb{R}^3 \rightarrow \mathbb{R}$:

$$f(x_1, x_2, x_3) = \left(1 + \sqrt{x_1} + \frac{1}{x_2} + \frac{1}{\sqrt{x_3}}\right)^2 + \eta.$$  \hspace{1cm} (57)

216 training data sets are generated with 10-fold cross-validation and 125 for tests; 2 gaussmtype2 IT2MFs for each input, 8 rules, and 50 epochs. Once the ANFIS and IT2FNN models are identified, a comparison is made with RMSE statistic values and 10-fold cross-validation. Table 2 and Figure 8 show the resultant RMSE (CHK) values for ANFIS and IT2FNN. It can be seen that IT2FNN architectures [31] perform better than ANFIS.

Experiment 3. Predicting the Mackey-Glass chaotic time series.

Mackey-Glass chaotic time series is a well-known benchmark [39] for systems modeling and is described as follows:

$$\dot{x}(t) = \frac{0.1x(t - \tau)}{1 + x^{10}(t - \tau)} - 0.1x(t).$$  \hspace{1cm} (58)

1200 data sets are generated based on initial conditions $x(0) = 1.2$ and $\tau = 60$, using fourth order Runge-Kutta method adding different levels of uniform noise. For comparing with other methods, an input-output vector is chosen for IT2FNN model with the following format:

$$[x(t-18), x(t-12), x(t-6), x(t); x(t+6)].$$  \hspace{1cm} (59)

Four-input and one-output IT2FNN model is used for Mackey-Glass chaotic time series prediction, choosing 500 data sets for training and 500 test data data sets with 10-fold cross-validation test, 2 IT2MFs for each input with membership function igaussmtype2, 16 rules, and 50 epochs. ANFIS and IT2FNN models are identified, comparing RMSE statistical values with 10-fold cross-validation. Table 3 and Figures 9 and 10 show the number of $\gamma$ points out of uncertainty interval $\tilde{Y}(x) \in [\tilde{y}_l(x), \tilde{y}_r(x)]$ evaluated by IT2FNN model, RMSE training values (TRN) and test (CHK) obtained for ANFIS and IT2FNN models. It can be seen that IT2FNN model architectures predict better Mackey-Glass chaotic time series.

5. Conclusions

In this paper we have shown that an interval type-2 fuzzy neural network (IT2FNN) is a universal approximator. Simulation results of nonlinear function identification using
Table 3: RMSE (TRN/CHK) and $\zeta$ values determined by ANFIS and IT2FNN models with 10 fold cross-validation for Mackey-Glass chaotic time series prediction with $\tau = 60$.

<table>
<thead>
<tr>
<th>SNR (dB)</th>
<th>ANFIS</th>
<th>IT2FNN-0</th>
<th>IT2FNN-1</th>
<th>IT2FNN-2</th>
<th>IT2FNN-3</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tau = 60$</td>
<td>TRN</td>
<td>CHK</td>
<td>$\zeta$</td>
<td>TRN</td>
<td>CHK</td>
</tr>
<tr>
<td>0</td>
<td>0.3403</td>
<td>0.3714</td>
<td>NA</td>
<td>0.2388</td>
<td>0.2687</td>
</tr>
<tr>
<td>10</td>
<td>0.1544</td>
<td>0.1714</td>
<td>NA</td>
<td>0.1312</td>
<td>0.1456</td>
</tr>
<tr>
<td>20</td>
<td>0.0929</td>
<td>0.1007</td>
<td>NA</td>
<td>0.0708</td>
<td>0.0781</td>
</tr>
<tr>
<td>30</td>
<td>0.0788</td>
<td>0.0847</td>
<td>NA</td>
<td>0.0526</td>
<td>0.0582</td>
</tr>
<tr>
<td>Free</td>
<td>0.0749</td>
<td>0.0799</td>
<td>NA</td>
<td>0.0414</td>
<td>0.0477</td>
</tr>
</tbody>
</table>

The student J. R. Castro was supported by a scholarship from MYCDI, UABC-CONACYT.

References


Submit your manuscripts at http://www.hindawi.com